CLAIMS

What is claimed is:

1. A method for scheduling multiple units of data requesting access to multiple ports in a network, the method comprising:

generating a request matrix that represents requests from particular units of data for particular ports;

generating a shuffle control that indicates a particular rearrangement of request matrix elements;

generating a shuffled request matrix, including;

rearranging, according the shuffle control, a set request matrix elements selected from a group comprising request matrix rows and request matrix columns; and

rearranging, according to a reversed shuffle control, a set of matrix elements comprising a member of the group that was not selected to be rearranged according to the shuffle control;

performing arbitration on the shuffled request matrix to generate a shuffled grant matrix that represents shuffled granted requests; and

generating a grant matrix, including applying a de-shuffle control to shuffled grant matrix elements including rows and columns.

2. The method of claim 1, wherein the multiple units of data are cells and the ports are egress ports of a packet switch, and wherein the method further comprises using the de-shuffled grant matrix to schedule a crossbar in the packet switch to perform cell transfers for one cell time.

3. The method of claim 2, wherein the rearranging according to the reversed shuffle control occurs at alternate cell times.

4. The method claim 3, wherein at cell times during which the rearrangement according to the reversed shuffle control does not occur, the request matrix rows and columns are each rearranged according to the shuffle control.

5. The method claim 1, wherein the shuffle control comprises a reassignment of positions among respective matrix elements, wherein the matrix elements include rows and columns, and wherein the reversed shuffle control indicates a reassignment of positions among the respective matrix elements that is the reverse of the shuffle control reassignment.

6. The method claim 5, further comprising generating the shuffle control using software, including:

performing a random_permute function to generate shuffle controls;

storing the shuffle controls in a random access memory ("RAM"); and

accessing the generated shuffle controls in sequence to generate shuffled request matrices.

7. The method claim 5, further comprising generating the shuffle controls using at least one pseudo-random number generator.

8. The method claim 5, further comprising deterministically generating the shuffle controls.

9. The method claim 1, wherein the arbiter is a wrapped wavefront arbiter ("WWFA").

10. A switch fabric, comprising:

a plurality of ingress ports;

a plurality of egress ports;

a crossbar selectively configurable to couple ingress ports to egress ports;

a scheduler coupled to the ingress ports, the egress ports, and the crossbar, the scheduler comprising,

a shuffle component that receives a shuffle control value that indicates a particular rearrangement of request matrix elements, wherein a request matrix represents requests from particular ingress ports for particular egress ports, and wherein the shuffle control component generates a shuffled request matrix, including,

rearranging, according to the shuffle control value, a set of request matrix elements selected from a group comprising request matrix rows and request matrix columns; and

rearranging, according to a reversed shuffle control value, a set of matrix elements comprising a member of the group that was not selected to be rearranged according to the shuffle control value;

performing arbitration on the shuffled request matrix to generate a shuffled grant matrix that represents shuffled granted requests; and

a de-shuffle component that generates a grant matrix, including applying a de-shuffle control value to shuffled grant matrix elements including rows and columns; wherein the grant matrix is used to configure the crossbar.

11. The switch fabric of claim 10, further comprising a shuffle/de-shuffle control component coupled to the shuffle component and to the de-shuffle component, wherein the shuffle/de-shuffle control component generates control signals under software direction from a central processing unit interface to configure the crossbar to perform data cell transfers from ingress ports to egress ports once each cell time.

12. The switch fabric of claim 11, wherein the rearranging according to the reversed shuffle control value occurs at alternate cell times.

13. The switch fabric of claim 12, wherein at cell times during which the rearrangement according to the reversed shuffle control value does not occur, the request matrix rows and columns are each rearranged according to the shuffle control value.

14. A method for scheduling data through a component in a network, the method comprising:

allocating egress port bandwidth for each of a plurality of egress ports to various inputs;

assigning credits to each of the various inputs in proportion to a predetermined bandwidth allocation for an egress port;

if an input requests access to an egress port and the input has at least one credit for the requested egress port, allowing the request to proceed to an arbiter; and

when an input receives a grant of access to a requested egress port from the arbiter, decrementing the credits of the input for the egress port by one.

15. The method of claim 14, further comprising:

if an input has zero credits for an egress port, disallowing any requests from the input for the egress port from proceeding to the arbiter; and

when all of the inputs have zero credits for the egress port, resetting the credits, comprising reassigning credits to each of the various inputs in proportion to the predetermined bandwidth allocation for the egress port.

16. The method of claim 14, further comprising, when an input has a request for an egress port, the input has credits ≤ zero for the requested egress port, and no other inputs have pending requests for the egress port, allowing the request to proceed to the arbiter and decrementing the credits of the input for the egress port by one.

17. The method of claim 16, further comprising considering a priority assignment in allowing a request to proceed to an arbiter, including:

assigning a first priority to requests from inputs that have credits > zero for the requested egress port; and

assigning a second priority to requests from inputs that have credits ≤ zero for the requested egress port, wherein second priority requests are only granted when no first priority requests are pending.

18. The method of claim 16, further comprising, when all of the inputs have credits ≤ zero for the egress port, updating the credits, comprising adding credits in proportion to the predetermined bandwidth allocation to each of the various inputs.

19. The method of claim 16, further comprising a maximum negative value, wherein the method further comprises, when an input has credits = (maximum negative value) for an egress port, disallowing any requests from the input for the egress port from proceeding to the arbiter.

20. The method of claim 19, further comprising, when all of the inputs have (maximum negative value) ≤ current credits ≤ zero for an egress port, updating the credits, comprising adding credits to the current credits for each of the various inputs in proportion to the bandwidth allocation for the egress port.

21. The method of claim 14, wherein the component comprises a packet switch, and the various inputs comprise a plurality of ingress ports in the packet switch, and wherein each egress port of the packet switch individually allocates bandwidth among the ingress ports.

22. The method of claim 14, wherein the component comprises an input queued with virtual output queuing ("IQ with VOQ") packet switch with a plurality of ingress ports such that each ingress port of the component comprises a virtual output queue for each egress port, and wherein the various inputs comprise the virtual output queues.

23. The method of claim 14, wherein the component comprises an input queued with virtual output queuing ("IQ with VOQ") packet switch with a plurality of ingress ports such that each ingress port of the component comprises a plurality of virtual output queues, and wherein each of the virtual output queues corresponds to a combination of an egress port and at least one item selected from a group comprising a data class and a data priority.

24. An apparatus for scheduling data through a network component, the apparatus comprising:

a plurality of component ingress ports, each comprising a plurality of ingress port queues;

a plurality of ingress port processors, each receiving requests for access to multiple component egress ports from a plurality of ingress port queues, wherein an egress port processor includes,

credit update circuitry for receiving an initial number of credits for each queue, wherein the initial number of credits for a queue corresponds to an allocation of bandwidth by one egress port to one queue;

request processing circuitry coupled to the credit update circuitry and coupled to receive a request from a queue for access to an egress port, wherein the request processing circuitry determines whether to allow the request to proceed to an arbiter based on criteria including whether a requesting queue's number of credits is greater than a predetermined saturation value.

25. The apparatus of claim 24, wherein the apparatus is cooperative with a strict priority scheme that assigns data one of a plurality of priorities, and wherein all data on the ingress ports is assigned a same priority for purposes of determining whether to allow a request to proceed to the arbiter.

26. The apparatus of claim 24, wherein the apparatus is cooperative with a strict priority assignment scheme that assigns data one of a plurality of priorities, and wherein all of the data on the ingress ports is initially assigned one priority for purposes of determining whether to allow a request to proceed to an arbiter, and when a requesting queue's number of credits is equal to or less than zero, the requesting queue is assigned a different priority that is lower than the initially assigned priority, such that the requesting queue's request is allowed to proceed to the arbiter if no other queue with the initially assigned priority has a pending request for the egress port.

27. The apparatus claim 24, further comprising:

grant allocation circuitry that receives a grant from the arbiter granting access to an egress port and allocates the grant to one of a plurality of data classes according to a predetermined allocation scheme;

request update circuitry coupled to the grant allocation circuitry for receiving the allocated grant and new requests; and

request count circuitry coupled to the request update circuitry for receiving the allocated grant and new requests and updating request counts for respective classes of data accordingly.

28. The apparatus of claim 27, wherein the credit update circuitry is further coupled to receive the allocated grant and, in response, decrement a number of credits for a queue that was allocated the grant.

29. The apparatus of claim 28, wherein:

the request processing circuitry is coupled to the credit update circuitry to receive current credit values for all of the queues;

the request processing circuitry is coupled to the egress ports to send a flow_done signal to each egress port to indicate that all queues for a respective ingress port have exhausted their allocations of that egress port's bandwidth; and

the request processing circuitry receives an egress_done signal from each egress port indicating that the respective egress port has no pending requests from any ingress ports.

30. The apparatus of claim 29, wherein the credit update circuitry responds to the egress_done signal by resetting credits for each queue to the initial number.

31. A method for scheduling data through a network component in a network that uses a strict priority scheme, the method comprising:

allocating egress port bandwidth for each of a plurality of component egress ports to various component ingress ports in a weighted round robin manner, wherein the allocation includes assigning credits to each of the various ingress ports in proportion to a bandwidth allocation for an egress port;

determining which pending requests from ingress ports for egress ports will be passed to a crossbar scheduler, wherein the determination depends on a current number of credits assigned to an ingress port and a current strict priority assigned to the ingress port;

passing requests to the crossbar scheduler in the form of a request matrix;

operating on the request matrix, including,

generating a shuffled request matrix using the crossbar scheduler, including;

rearranging, according to a shuffle control value, a set of request matrix elements selected from a group comprising request matrix rows and request matrix columns; and

rearranging, according to a reversed shuffle control value, a set of matrix elements comprising a member of the group that was not selected to be rearranged according to the shuffle control value;

performing arbitration on the shuffled request matrix using to generate a shuffled grant matrix that represents shuffled granted requests;

generating a grant matrix, including applying a de-shuffle control value to shuffled grant matrix elements including rows and columns; and

using the grant matrix to configure the crossbar.

32. The method of claim 31, wherein allocation occurs at at least two levels, including:

a first level at which bandwidth is allocated among the ingress ports by a single egress port;

a second level at which bandwidth is allocated among multiple flows within each of the ingress ports, wherein a flow is characterized by an ingress port, an egress port, and a data class;

a third level at which bandwidth is allocated among items selected from a group comprising at least one sub-port and at least one data sub-class.

33. The method of claim 32, wherein multiple data classes are mapped to a single strict priority.

34. The method of claim 31, wherein:

all flows are initially assigned an initial number of credits in proportion to bandwidth allocated to the flow by an egress port, and all flows are initially assigned a same strict priority, and

a flow's request for an egress port is passed to the scheduler when the flow has a credit balance for the egress port that is greater than zero.

35. The method of claim 34, wherein all flows are reassigned the initial number of credits for an egress port when all flows have credit balances of zero for the egress port.

36. The method of claim 31, wherein:

all flows are initially assigned an initial number of credits in proportion to bandwidth allocated to the queue by an egress port, and all flows are initially assigned a same strict priority; and

if a flow has zero credits for the egress port, the flow is assigned a different strict priority that is lower than the initially assigned strict priority such that requests from the flow for the egress port may be passed to the scheduler if no flows with higher priority have pending requests for the egress port.

37. The method of claim 36, further comprising a saturation number of credits, which is a negative number such that if a flow has the saturation number of credits for an egress port, no requests from the flow for the egress port will be passed to the crossbar scheduler.

38. The method of claim 37, wherein, when all flows have the saturation credit number for an egress port, all flows are reassigned the initial numbers of credits and the initial same strict priority.

39. The method of claim 31, wherein the rearranging according to the reversed shuffle control occurs every other time the crossbar is configures.

40. The method of claim 39, wherein when the rearrangement according to the reversed shuffle control does not occur, the request matrix rows and columns are each rearranged according to the shuffle control value.

41. The method of claim 40, wherein when the shuffle control value indicates a reassignment of positions among respective matrix elements, wherein the matrix elements include rows and columns, and wherein the reversed shuffle control value indicates a reassignment of positions among the respective matrix elements that is the reverse of the reassignment indicated by the shuffle control value.

42. A method, comprising:

generating a plurality of values, in the form of a matrix, representing a plurality of requests to transfer a plurality of data between a plurality of ingress ports and a plurality of egress ports;

generating a random series of numbers representing matrix elements selected from a group comprising matrix rows and matrix columns;

rearranging, responsive to the random series of numbers, a set of matrix elements selected from the group; and,

rearranging, responsive to a reverse random series of numbers, a set of matrix elements comprising a member of the group that was not selected to be rearranged responsive to the random series of numbers.

43. The method of claim 42, wherein the plurality of data are cells and the plurality of egress ports are a plurality of egress ports of a packet switch.

44. The method of claim 42, wherein the rearranging according to the reversed random series of numbers occurs at alternate cell times.

45. A method, comprising:

assigning a plurality of credit values to each of a respective plurality of inputs in proportion to a predetermined bandwidth allocation for an egress port;

determining whether an input in the plurality of inputs requests access to the egress port;

determining whether a credit value associated with the input is greater than a predetermined threshold value;

allowing the request to proceed to an arbiter responsive to the determining steps; and

decrementing the credit value of the input for the egress port responsive to the input receiving a grant of access to the egress port from the arbiter.

46. The method of claim 45, wherein the threshold value is zero; and the method further comprises the step of:

reassigning the plurality of credit values to each of the respective plurality of inputs in proportion to the predetermined bandwidth allocation for the egress port responsive to the plurality of credit values being zero.

47. The method of claim 45, wherein the allowing step further includes allowing the request to proceed when no other inputs have pending requests for the egress port.